WWW.SCIAM-MAGAZINE.NAROD.RU

Программа, играющая в шашки, часто «видит» игру на ход дальше противника

А. К. ДЬЮДНИ

МОЖНО ли написать такую программу, играющую в шашки, которая никогда не проиграет человеку, никогда-никогда?» Мальчик был лет десяти и взглянул на нас, оторвавшись от терминала, с таким глубокомысленным, но невинным выражением, которое свойственно лишь очень юным. Он играл в шашки с программой, написанной моими студентами. Программа была настроена так, чтобы не просчитывать развитие игры на много ходов вперед, и паренек играл с ней довольно успешно.

Этот вопрос был задан мальчиком на выставке, устроенной на нашем факультете несколько лет назад, и на него можно ответить по-разному в зависимости от того, о каком варианте игры в шашки идет речь. Вопрос имеет также непосредственное отношение к важнейшим проблемам создания искусственного интеллекта, в частности к программированию «разумного» поведения компьютерных программ. Об этих проблемах мы поговорим несколько позже.

Когда мальчик задал нам свой вопрос, он играл в шашки на доске с числом клеток, равным 6 × 6. Это лишь один вариант из бесконечного многообразия шашечных игр. В настоящей статье мы рассмотрим варианты игры 4 × 4, 6 × 6 и 8 × 8 (классические шашки) и обсудим перспективы применения программ, написанных для этих игр. Вообще говоря, можно определить шашки n-го порядка, т.е. игру на доске с числом клеток $2n \times 2n$. В каждом случае шашки, или шашечные фигуры, в исходном положении занимают все темные клетки, за исключением двух рядов в средней части доски. Правила игры в шашки n-го порядка более подробно рассмотрены на с. 108.

В случае варианта 4 × 4 на вопрос, заданный мальчиком, можно, по-видимому, дать положительный ответ. Для варианта 6 × 6 ответом будет «очень вероятно», а для игры 8 × 8 — «возможно». Мои ответы основаны на сочетании личного опыта, литературных данных и просто интуиции.

РАССМОТРИМ вариант игры 4 × 4. В этом случае у каждого играющего по две фигуры и игра редко продолжается более 10 ходов, по крайней мере когда играют квалифицированные игроки. Новички могут сделать 20 ходов или даже больше, прежде чем ничейный результат станет для них очевидным. Кстати, если уж речь зашла о ничьих, то в одном из последних турниров по шашкам 4 × 4 в городе Шашигрсбурге было зафиксировано шесть ничьих подряд и лишь в седьмой партии победа по решению судейской коллегии была присуждена гроссмейстеру Самюэлю Енсену Траскотту — его противник дважды случайно дотронулся до фигуры, которой нельзя было ходить.

Во всяком случае, вариант игры 4 × 4 можно проанализировать вручную. По-видимому, при разумной игре закономерным будет ничейный результат, поэтому игрок, проявляющий достаточно внимания, никогда не должен проигрывать. То же можно сказать и о достаточно хорошей программе.

В варианте игры 6 × 6 каждый партнер имеет шесть фигур и в среднем игра завершается после примерно 20 ходов. Если программу, написанную нашими студентами, настроить на максимально возможное просматривание игры вперед, то ей требуется от пяти до десяти минут для выбора очередного хода. При такой настройке программа играла очень сильно и казалась совершенно неуязвимой вплоть до эндшпиля, к которому она подходила с таким большим преимуществом в фигурах или в позиции, что обычно доводила партию до победы или в крайнем случае сводила ее вничью. Однако время от времени кто-нибудь из нас сохранял достаточно фигур, чтобы победить программу в эндшпиле, который она играла удивительно слабо.

В варианте игры 8 × 8 каждый партнер имеет по 12 фигур, и, когда играют квалифицированные игроки, число ходов достигает 30 и более. На турнирах по стандартным шашкам теперь стало уже обычной практикой устанавливать первый ход по жребию (имеется в виду не очередность, а вариант хода). Такой обычай был принят в Англии в начале века, потому что мастера этой игры почти всегда начинали со своего излюбленного хода с хорошо изученным продолжением, что приводило ко все возрастающему числу ничейных результатов. Можно ли объяснить изобилие ничьих тем, что шашки по своей природе являются ничейной игрой, или все дело в консерватизме мастеров, трудно сказать. Как бы то ни было, шашки в варианте игры на доске 8 × 8 остаются увлекательной и трудной игрой. С математической точки зрения шашки несколько уступают по сложности шахматам, однако есть люди, одинаково хорошо играющие в обе эти игры и не считающие шахматы игрой более высокого уровня.

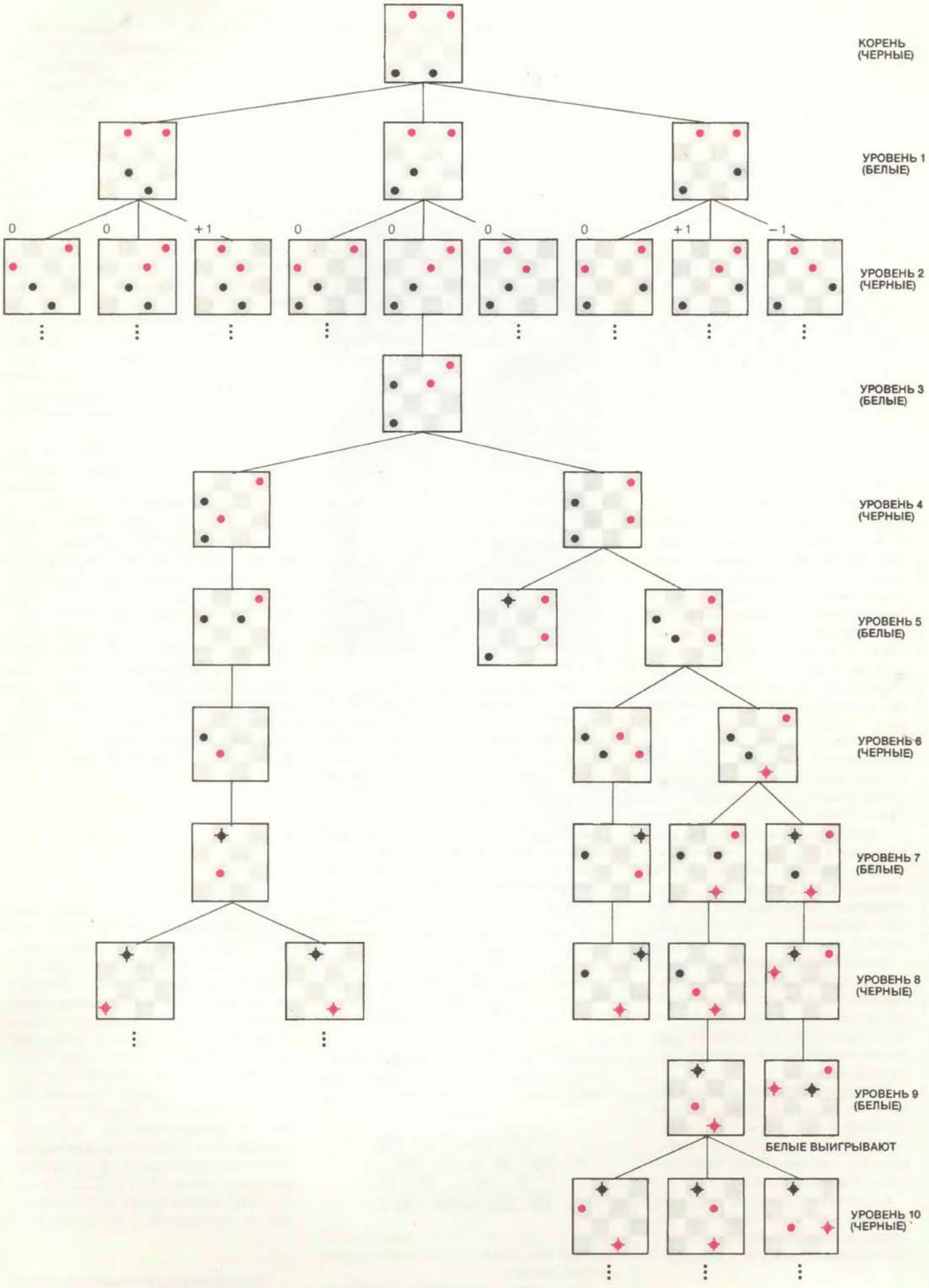
Две наиболее известные программы, играющие в шашки, были созданы А. Самюэлем из фирмы IBM (International Business Machines Corporation) в 60-х годах и Э. Енсеном и Т. Траскоттом из Университета Дьюка в 1977 г. В 1962 г. шашечная программа Самюэля победила Роберта Нили, в прошлом чемпиона шт. Коннектикут. Хотя эта программа играет очень сильно, она все же уступает игре гроссмейстерского уровня. Программа Енсена — Траскотта еще сильнее и по игре находится на уровне ведущих мастеров мира.

В 1979 г. программа Енсена — Траскотта играла против Элберта Лаудера, считающегося одним из лучших игроков в США. Из пяти партий две закончились вничью, одну программа выиграла и две проиграла. Проигрыши, по-видимому, объясняются некоторыми недочетами, присущими программе в эндшпиле: когда Лаудер понял, что две шашки противника оставались «на страже» на задней линии, он воспользовался слабостью этой позиции и выиграл последние две партии. Следует, однако, заметить, что, играя против программы Енсена — Траскотта, Лаудер экспериментировал и играл несколько рискованно. Его проигрыш в одной из партий объясняется, по всей видимости, попыткой «заманить» программу на необычные позиции. Возможно, этого не следовало делать в игре с программой такого высокого уровня.

Тем временем Марион Ф. Тинзли, чемпион мира по шашкам, с интересом наблюдает за событиями из своего дома в Теллахассе (шт. Флорида). Я не знаю, играл ли Тинзли когда-нибудь с лучшими шашечными программами. Однако известно, что две лучшие программы играли между собой. В двух партиях против программы Самюэля программа Енсена — Траскотта оказалась, по словам Траскотта, «на голову выше» и выиграла их.

В БОЛЬШИНСТВЕ удачно играющих программ можно выделить три основные компоненты: генератор ходов, функцию, оценивающую позицию, и минимаксную процедуру. Еще один важный элемент — это «дерево» игры; оно участвует в работе всех трех основных компонент игровой программы. Генератор ходов строит дерево; к оценивающей функции происходит обра-

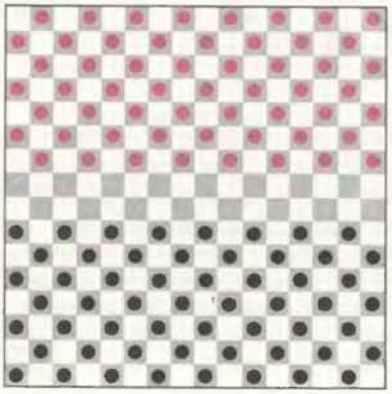
WWW.SCIAM-MAGAZINE.NAROD.RU



Дерево игры в шашки для варианта 4 × 4. Полностью представлены первые два хода и некоторые варианты продолжений

WWW.SCIAM-MAGAZINE.NAROD.RU

Правила игры в шашки n-го порядка



На доске, представляющей собой квадрат размерами $2n \times 2n$ клеток, попеременно светлых и темных, каждый игрок располагает $n^2 - n$ шашек на темных клетках на противоположных сторонах доски. Два средних ряда клеток остаются пустыми. У каждой стороны есть «одиночный угол» (темная клетка в левом углу доски), а также «двойной угол» (справа). Черные ходят первыми.

Простая шашка может ходить по диагонали вперед на прилегающую незанятую темную клетку. Если эта клетка занята фигурой противника, а следующая за ней по диагонали клетка свободна, то шашка может перескочить фигуру противника, причем последняя считается захваченной и убирается с доски. Любая возможность такого захвата используется в обязательном порядке; если одновременно возможны два или более захватов, по крайней мере один из них должен быть сделан.

Когда простая шашка достигает последнего ряда на противоположной стороне доски, она становится дамкой. Дамка может ходить как вперед, так и назад (тоже по диагонали) и может бить в обоих направлениях.

Игра заканчивается победой одного из игроков, когда другой игрок не может сделать ни одного хода. Если оба игрока приходят к выводу, что выигрыш невозможен, объявляется ничья.

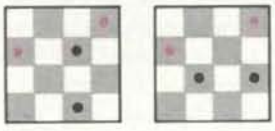
На рисунке показана исходная позиция для игры восьмого порядка на доске размерами 16 × 6 с 56 фигурами с каждой стороны.

щение в конце каждой ветви дерева, а минимаксная процедура применяется ко всему дереву в целом.

Корневой узел дерева представляет собой исходную позицию на доске, а каждая ветвь ведет к какой-то позиции, достигаемой при помощи одного хода. Примечательно, что подобное дерево изображается обычно «вверх ногами» — его корень располагается вверху. Дерево можно «разделить» на уровни так, что все позиции n-го уровня соответствуют всем возможным ходам одного игрока, делающего n-й ход. Хотя игровые деревья, как правило, не очень высоки (точнее говоря, глубоки), они могут быть очень кустистыми. Дело в том, что число позиций на каждом уровне более чем вдвое превышает число позиций на предыдущем уровне. Даже если у каждой позиции на доске было лишь две последующие позиции, то число ветвей очень скоро достигло бы огромной величины. Дерево, разветвившееся подобным образом 64 раза, имело бы больше ветвей, чем все деревья, растущие на Земле, вместе взятые. (В природе деревья ветвятся не более 12 раз от корня до листьев.) Из-за того что число ветвей игрового дерева возрастает по экспоненциальному закону, т.е. имеет место «экспоненциальный взрыв», компьютер может исследовать лишь небольшую часть продолжений в такой игре, как шашки на доске 8 × 8.

Дерево шашечной игры создается генератором ходов — программой, которой в качестве входных данных задается текущая позиция на доске, а также указание о том, кто из противников должен сделать очередной ход. Выходом генератора ходов являются все возможные позиции, достижимые из заданной позиции после одного хода. Рассмотрим следующую позицию на доске 4 × 4, предполагая, что ходить должны черные фигуры:

Для черных имеется два возможных хода, и генератор ходов выдает две позиции:



Затем генератор ходов может применяться уже к каждой из этих позиций в отдельности.

Читателям, желающим разобраться, как работает генератор ходов, мы предлагаем достроить дерево игры на доске 4 × 4, отправляясь от рассмотренного нами уровня. Например, на следующем уровне будет только две позиции, поскольку в каждом случае правила игры вынуждают белых* бить фигуру противника. На уровне, следующем за этим, будет опять две позиции, затем три (в одной из них белые выигрывают), затем 2, 5, 10 и, наконец, 24. Тем, кто пожелает последовать еще дальше, мы дадим два совета. Во-первых, нужно просматривать каждый уровень в поисках идентичных позиций и строить продолжение лишь для одной из таких позиций. Во-вторых, для каждой новой позиции нужно просматривать дерево снизу вверх, чтобы найти идентичную позицию. Если эта позиция уже встречалась в игре, то на повторно встретившейся позиции ветвь нужно закончить, пометив ее конец как «ничью».

Как работает программа, генерирующая ходы? Первый вопрос заключается в том, каким образом позиции на доске представляются в памяти компьютера. В качестве простейшей схемы можно принять матрицу — двумерный массив, элементы которого соответствуют каждой клетке шашечной доски. Например, пустой клетке соответствует 0, клетке, занимаемой «простой» шашкой, — 1, а клетке, которую занимает «дамка», — 2. Знак числа может отражать принадлежность фигуры к черным или белым, плюс — для черных и минус — для белых фигур. Первая позиция на доске 4 × 4, показанная выше, может быть представлена матрицей следующего вида:

$$\begin{matrix} 0 & 0 & 0 & -1 \\ -1 & 0 & 0 & 0 \\ 0 & +1 & 0 & 0 \\ 0 & 0 & +1 & 0 \end{matrix}$$

Ходы, разрешенные правилами игры для каждой фигуры, зависят от того, какой стороне она принадлежит, от ее позиции, а также от того, заняты или свободны прилежащие по диагонали клетки. В рассматриваемом случае генератор ходов сканирует массив строку за строкой, пока не наталкивается на положительный элемент (фигура черных) в строке 3 колонки 2; мы будем обозначать такую позицию как (3,2). Поскольку ход принадлежит черным и фигура является «простой» шашкой, а не «дамкой» (значение элемента матрицы равно 1), программа исследует клетки (2,1) и (2,3). Клетка (2,1) уже занята фигурой (и перескочить ее невозможно), но клетка (2,3)

* На рисунках для наглядности эти фигуры изображены красным цветом. — *Прим. перев.*

WWW.SCIAM-MAGAZINE.NAROD.RU

Издательство **МИР** *предлагает:*

Гааль Э., Медьеши Г., Верецкеи Л.

ЭЛЕКТРОФОРЕЗ В РАЗДЕЛЕНИИ БИОЛОГИЧЕСКИХ МАКРОМОЛЕКУЛ

Перевод с английского

Новая книга венгерских авторов посвящена методу, позволяющему разделять макромолекулярные вещества и исследовать каждое из них в отдельности. Рассмотрены теоретические основы электрофореза на бумаге, в колонках и блоках, в гелях, обработка электрофореграмм, особенности электрофореза белков, нуклеопротеидов, нуклеиновых кислот, гликозаминогликанов, варианты иммуноэлектрофореза. Обсуждаются следующие вопросы: теоретические и технические основы методов электрофореза; аналитический электрофорез с движущейся границей в свободной среде; зональный электрофорез на фильтровальной бумаге, других поддерживающих средах и на ацетате целлюлозы; электрофорез в колонках, горизонтальных и вертикальных блоках; электрофорез в гелях (агаровом, агарозном, крахмальном, полиакриламидном); изоэлектрофокусирование и изотахофорез; методы обработки электрофореграмм для выявления зон разделенных макромолекул; электрофорез и иммуноэлектрофорез белков и ферментов; электрофорез нуклеиновых кислот, нуклеопротеидов, полинуклеотидов, гликозаминогликанов.

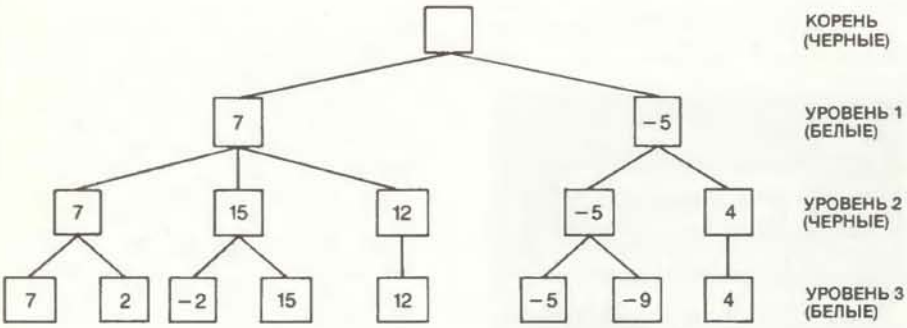
Книга предназначена для биологов всех специальностей, пользующихся методами современной биохимии, для врачей-лаборантов, а также для студентов и преподавателей университетов, медицинских, педагогических и сельскохозяйственных институтов.

ЗАКАЗЫ НА ЭТУ КНИГУ НАПРАВЛЯЙТЕ ПО АДРЕСУ: 129820 Москва, И-110, ГСП, 1-й Рижский пер., 2, издательство «Мир», отдел распространения. Организациям следует заверять свой заказ печатью. Индивидуальным заказчикам книга будет выслана наложенным платежом.

1982, 30 л. Цена 3 р. 20 к.

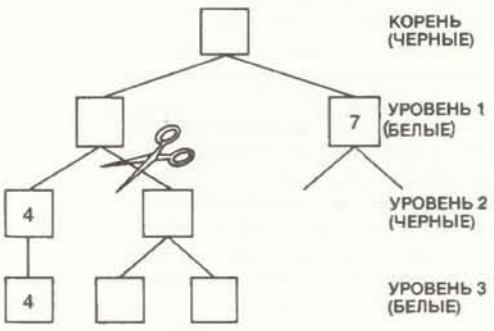
WWW.SCIAM-MAGAZINE.NAROD.RU

Минимаксная процедура и усечение дерева



Минимаксная процедура применяется к игровому дереву после того, как все дозволенные правилами ходы вплоть до установленной глубины были сгенерированы и при помощи оценивающей функции получены оценки позиций на конце каждой ветви. Здесь ход принадлежит черным, дерево было просмотрено на глубину в три уровня и оценивающая функция присвоила каждой конечной позиции целое число, характеризующее относительную ценность позиции для черных. Задача минимаксной процедуры — выбрать для черных ход на основе этих оценок.

Заметим, что наиболее выгодная последовательность ходов для черных проходит через квадратики, помеченные цифрами 7, 15 и 15. Для белых предпочтительнее последовательность −5, −5, −9. Минимаксная процедура учитывает эти противоположные устремления. Программа сначала анализирует все ходы третьего уровня, являющиеся «потомками» определенного хода — «родителя» на втором уровне. Поскольку на втором уровне должны ходить черные, «родителю» присваивается значение наиболее успешного «потомка». Та же процедура используется, чтобы присвоить значения позициям первого уровня, но поскольку здесь ход принадлежит белым, выбираются наименьшие значения. Если оба игрока будут оценивать позиции по этому методу, то игра пойдет по пути 7, 7 и 7.



Игровые деревья глубиной больше трех уровней могут достигать чудовищных размеров, и поиск на них можно проводить только с использованием усечения. Здесь ход принадлежит черным, и при помощи частичного поиска уже была получена оценка 7 для одного из ходов на первом уровне. При анализе другого возможного хода программа выяснила, что при одном из возможных ответов белых на уровне 2 оценка позиции равна 4. Поэтому нет необходимости исследовать оставшуюся часть дерева — поскольку белые выберут ход с наименьшей оценкой, при альтернативном выборе черных на уровне 1 оценка не может иметь значения, большего 4, и поэтому ход с оценкой 7 предпочтительнее. Исключение путей, которые черные могут игнорировать, называется «альфа-усечением», а соответствующий процесс для белых — «бета-усечением».

свободна. Поэтому программа строит новый массив, в котором +1 удаляется с позиции (3,2) и помещается в позицию (2,3):

```
 0  0  0−1
−1  0+1  0
 0  0  0  0
 0  0+1  0
```

Затем генератор ходов продолжает сканировать массив, пока не найдет следующий положительный элемент.

Чтобы следить за различными позициями на доске и соответствующими массивами, их можно последовательно пронумеровать в порядке генерации. Тогда состояние любой клетки в каждый момент игры будет определяться номером позиции на доске, а также номерами строки и колонки массива. По мере того как генератор ходов строит дерево игры, он должен записывать, в каком порядке позиции следуют друг за другом. Когда программа выбирает очередной ход, она просматривает дерево на несколько уровней вниз, затем вверх, затем снова вниз и т.д. Чтобы перемещаться по дереву таким образом, программа должна знать, какие позиции на доске являются предшествующими, или «родителями», а какие — последующими, или «потомками». Для этого у программы есть специальный массив, называемый таблицей указателей. Указателем для каждой позиции является список чисел, в котором первое число указывает на позицию, являющуюся «родителем» данной, а остальные числа — на ее «потомков».

СЛЕДУЮЩАЯ важная компонента программы, играющей в шашки, — это функция, оценивающая позиции. В качестве входной информации ей дается позиция на доске, и она выдает число, отражающее ценность этой позиции для черных фигур. Большое число означает, что если данная позиция будет достигнута в игре, то у черных будут хорошие шансы на выигрыш. Построить хорошую оценивающую функцию очень трудно, потому что при оценке позиции на доске нужно принимать во внимание слишком много факторов. Кроме того, в большинстве игр, включая шашки, не всегда ясно, какой вес приписать тому или иному фактору. Перечислим кратко простейшие из этих факторов:

Простые шашки. Прибавить 1 для каждой черной шашки на доске; вычесть 1 для каждой белой шашки.

Дамки. Прибавить 1 для каждой черной дамки; вычесть 1 для каждой белой дамки.

Центр. Прибавить 1 для каждой черной шашки, занимающей одну из четырех центральных клеток; соответст-

WWW.SCIAM-MAGAZINE.NAROD.RU

венно вычесть 1 для каждой черной шашки.

Продвижение. Прибавить 1 для каждой черной шашки; находящейся во втором ряду, 2 для каждой черной шашки в третьем ряду, 3 для каждой черной шашки (дамки) в четвертом ряду; вычесть соответствующие величины для белых фигур.

Подвижность. Прибавить 1 на каждый возможный ход для черных и вычесть 1 на каждый возможный ход для белых фигур.

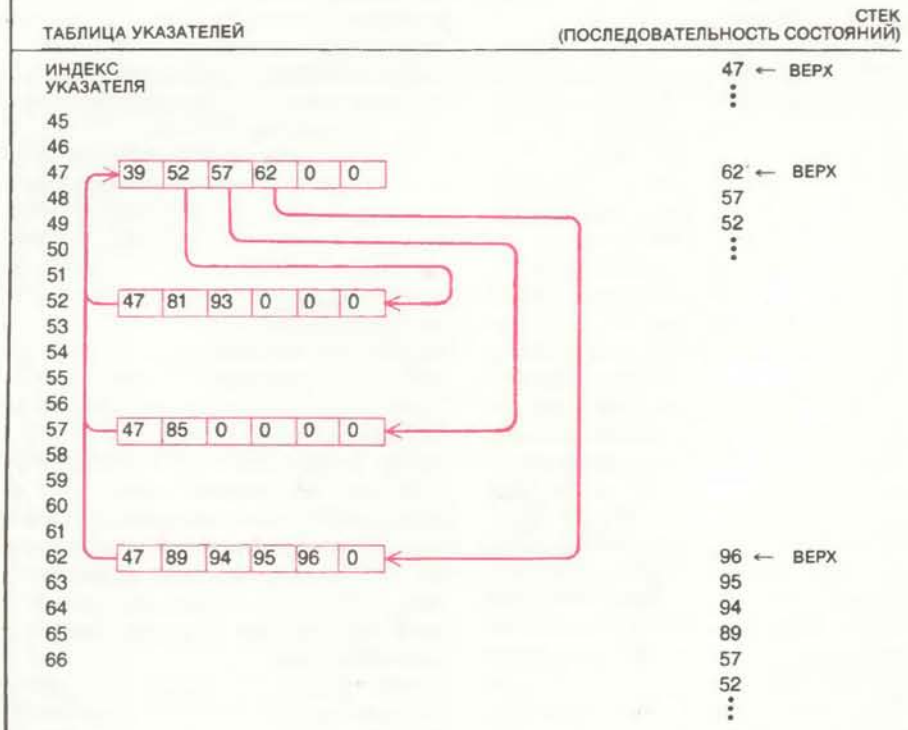
Каково относительное значение перечисленных факторов? Следует ли приписать дамкам вдвое больший вес, чем простым шашкам? Входит ли в «продвижение» информация, содержащаяся в таких факторах, как «шашки» и «дамки»?

Самюэль ответил на эти вопросы, позволив программе самой установить, какой вес приписать тому или иному фактору на основе игрового опыта. Оперируя 39 факторами (многие из которых были предложены знатоками игры или взяты из литературных источников), оценивающая функция Самюэля вначале приписала им один и тот же вес и просто складывала одни и те же числа. Полученная таким образом сумма характеризовала «ценность» позиции. Каждую проигранную партию программа анализировала, чтобы выявить фактор, который привел к поражению. В результате вес этого фактора уменьшался. Процесс обучения был ускорен за счет автоматизации анализа и многократных игровых испытаний.

Процедуры, оценивающие позицию, никогда не бывают совершенными. Если бы это было не так, то остальные компоненты игровой программы едва ли потребовались бы. Программе нужно было бы лишь исследовать позиции, достигаемые из данной при одном возможном ходе, оценить каждую из этих позиций и выбрать лучший ход. Существует своего рода компромисс между точностью оценивающей функции и числом уровней дерева, которые программа должна просматривать вперед. Если бы оценивающая функция могла лишь распознавать позиции, соответствующие окончательному выигрышу или проигрышу, то игровое дерево нужно было бы просматривать на всю глубину!

Третья главная компонента программы — минимаксная процедура, представляющая собой программу поиска на игровом дереве, которая пользуется оценками позиций на нижнем уровне, чтобы придать оценочные значения позициям, расположенным на более высоких (т.е. начальных) уровнях дерева игры. На какой-то стадии исследования дерева все позиции определенного уровня должны быть оценены этой функцией. Позиция *B* на уровне, непо-

Просмотр игрового дерева



При выборе очередного хода программа, играющая в шашки, должна проанализировать игровое дерево до какой-то установленной глубины, при этом каждая ветвь просматривается только один раз. При простом просмотре используется таблица указателей, где записываются отношения между ветвями, и структура данных, называемая стеком, которая следит за текущим положением программы на дереве и содержит список ветвей, не просмотренных еще программой.

Когда генерируется очередная позиция на доске, ей присваивается номер, служащий в качестве индекса в массиве указателей. Указатель, соответствующий позиции, сам представляет собой список номеров позиций. Первый номер в списке — это номер позиции-родителя, все остальные номера указывают ее потомков. Если программа достигла позиции 47, указатель говорит о том, что она должна продолжать просмотр через позиции 52, 57 и 62. Если достигнута позиция 62, программа должна вернуться к позиции-родителю — первый элемент указателя возвращает ее к позиции 47.

Стек должен следить за прохождением программы по таблице указателей. Стек может быть создан за счет организации массива STACK и переменной TOP, содержащей адрес верхнего элемента стека. Когда к стеку добавляется (вталкивается) или исключается (выталкивается) элемент, TOP увеличивается или уменьшается соответственно.

Элемент на верху стека — это номер позиции, которую нужно проанализировать следующей. Если на верху 47, алгоритм просмотра дерева выталкивает это значение из стека и затем анализирует список указателя для позиции 47. Первый номер в списке (указывающий на позицию-родителя по отношению к позиции 47) игнорируется, а оставшиеся ненулевые элементы вталкиваются в стек и переменная TOP получает соответствующее значение. Следующей должна анализироваться позиция 62.

Каждый раз, когда программа достигает максимально дозволенной глубины, происходит обращение к оценивающей функции. Затем минимаксная процедура может «посмотреть» в список указателя, чтобы определить позицию-родителя и присвоить ей временное минимальное или максимальное значение — оценку. При просмотре других ветвей, исходящих из этой позиции-родителя, минимаксная процедура может изменить значение оценки.

WWW.SCIAM-MAGAZINE.NAROD.RU

средственно выше оцененного, получает оценку по следующему простому правилу: если в позиции *B* ходить должны белые, то выбирается минимальное значение из оценок, полученных для «потомков» *B*; оно принимается в качестве оценки позиции *B*. Если же ходить должны черные, то нужно выбрать максимальное значение.

Не трудно понять, из каких соображений было выбрано это правило. Если ход принадлежит белым, то черные вполне могут предположить, что белые выберут ход, который приведет к позиции с минимальным значением оценки для черных. Точно так же, если ход принадлежит черным, то будет выбрана позиция с максимально возможной оценкой. Таким образом, минимаксная процедура начинает действовать с самого нижнего просмотренного уровня дерева и, пользуясь оценками, полученными для этого уровня оценивающей функцией, переносит их вверх по дереву. Переходя от какого-то уровня к следующему, расположенному выше, она поочередно выбирает минимальные и максимальные оценки, в зависимости от того, какой стороне принадлежит ход. В конце концов минимаксная процедура возвращается к текущей позиции и дает черным оценку для каждого возможного хода в этой позиции. Естественно, черные выбирают ход с самой высокой оценкой.

ПРОГРАММА, играющая в шашки, помимо рассмотренных трех основных компонент, имеет и другие компоненты. Она должна обеспечить взаимодействие с играющим человеком — принимать ходы с клавиатуры терминального устройства и печатать ответные ходы. Кроме того, она может показывать изображение текущей позиции на доске, представляя теми или иными символами фигуры обеих сторон. Во многих случаях программа позволяет играющему с ней человеку устанавливать ту «степень мастерства», которую программа должна проявлять в игре. Как правило, это числовой параметр, соответствующий глубине просмотра игры вперед, т.е. числу уровней дерева игры, которые программа должна исследовать, выбирая очередной ход.

Когда устанавливается большая глубина просмотра, программе требуется больше времени для выбора хода, потому что каждый лишний уровень дерева может удвоить время расчетов. Чтобы сократить поиск, который должна выполнять программа, в минимаксную процедуру может быть включен оригинальный механизм, получивший название альфа-бета-усечения. Принцип его действия показан в таблице, помещенной ниже. Альфа-бета-усечение можно применять практически на каждом уровне дерева, и с его помощью можно избежать огромного числа лишних процедур поиска и оценки.

Алгоритм игры в шашки не очень трудно запрограммировать. По мнению Траскотта, не обязательно располагать большой ЭВМ, чтобы реализовать успешно играющую программу. «Для разработки программ, играющих в шашки, вполне подходит и микрокомпьютер, — пишет он, — и программисты-любители, возможно, внесут весомый вклад в эту интересную и малоисследованную область».

Для читателей, обладающих некоторым опытом в программировании, вышесказанного будет вполне достаточно, чтобы построить и реализовать собственную программу для игры в шашки. Менее опытным, наверное, потребуется дополнительная информация. Построить генератор ходов и оценивающую функцию — задача в общем-то несложная, а вот о минимаксной процедуре следует поговорить подробнее. Каким образом происходит перемещение по игровому дереву вниз и вверх, как при этом запоминаются уже просмотренные позиции и те, которые еще нужно просмотреть? Некоторые советы по поводу того, как осуществлять перемещения по дереву, приведены в таблице на с. 111.

Если игровая программа составлена, написана и отлажена, ее недостатки становятся очевидными, когда с ней начинают работать, установив параметр глубины просмотра игры на максимальное значение. Время, затрачиваемое на выбор хода, может оказаться слишком большим, или может не хватить памяти, чтобы разместить дерево игры. Существует много приемов, позволяющих сократить время вычислений и объем требуемой памяти. Например, в нашем представлении шашечной доски двумерным массивом половина элементов (а именно элементы, представляющие светлые клетки) всегда остается нулевой. Можно представить шашечную доску одномерным массивом, исключив таким образом пустые клетки. Однако при таком представлении шашечной доски построить генератор ходов будет уже труднее.

Вопрос о том, можно ли записать программу игры в шашки, которая никогда не проиграет человеку, сводится по существу к вопросу о том, что представляют собой шашки по своей природе. Существует ли такая стратегия для шашек n-го порядка, при которой черные фигуры всегда выигрывают?* В каком случае успеха могут добиться белые? Является ли игра ничейной? Исчерпывающий анализ для шашек n-го порядка вручную можно провести только тогда, когда $n = 2$. Я анализировал вариант 4×4, строя дерево игры в среднем на глубину 10 ходов. На этом уровне обнаруживается несколько выигрышей для черных и такое же небольшое число проигрышей. Остальные позиции, по-видимому, ведут к ничейному исходу — дамки бесконечно «гоняются» друг за другом по тесному игровому полю 4×4. Мне не удалось строго доказать ничейный исход во всех случаях, но, если в мои вычисления не закралась ошибка, минимаксная процедура приписывает начальной позиции на доске нулевую оценку — это означает ничью.

А как обстоит дело с вариантом 6×6? Является ли он ничейным? Когда программа, написанная моими студентами, играла сама против себя при глубоком просмотре игры вперед, партии редко достигали какого-либо определенного исхода. Основываясь только на этом факте, можно предположить, что вариант 6×6 представляет собой ничейную игру. Разумеется, если оба варианта игры — 4×4 и 6×6 — оказываются ничейными, то «под подозрение» попадает и вариант 8×8. Возможно, кто-нибудь из читателей, проявив редкое сочетание умения и смелости, напишет программу, анализирующую дерево игры для варианта 6×6. Располагая мощной ЭВМ, такой проект, возможно, удастся осуществить.

В отношении варианта n-го порядка некоторые интересные результаты были получены теоретиками. Оказывается, труднорешаемой является следующая задача: «Определить, могут ли выиграть черные для произвольной, разрешенной правилами исходной позиции, занимаемой белыми и черными дамками и шашками на доске с числом клеток $2n \times 2n$». В 1978 г. А. Фрэнкелю из Научно-исследовательского института Вейцмана в Израиле, а также М. Гэри и Д. Джонсону из фирмы AT&T Bell Laboratories в США удалось доказать, что эта задача относится к классу задач, «труднорешаемых в смысле полиномиальной памяти». Задача, труднорешаемая в смысле полиномиальной памяти (*P*-space трудная), является еще более трудоемкой, чем так называемые *NP*-полные задачи, а последние, по мнению большинства теоретиков, по-видимому, не имеют общего практически реализуемого решения*. Другими словами, доказан следующий факт: если бы программа

---

* Предполагается, что черные ходят первыми. — *Прим. ред.*

* О труднорешаемых и *NP*-полных задачах речь шла в предыдущем номере журнала в статье «О макаронном компьютере и других аналоговых устройствах» («В мире науки», № 8, с. 98). Более подробно эти проблемы рассмотрены в книге М. Гэри и Д. Джонсона «Вычислительные машины и труднорешаемые задачи» («Мир, 1982 г.). — *Прим. перев.*

WWW.SCIAM-MAGAZINE.NAROD.RU

могла в совершенстве играть в шашки n-го порядка, то время, затрачиваемое ею в среднем на один ход, возрастало бы быстрее, чем любая полиномиальная функция с увеличением n.

Вопрос о создании программы, играющей в шашки без проигрыша, в значительной мере связан с доступными вычислительными мощностями компьютеров. Даже если шашки по самой своей природе представляют собой ничейную игру, успешно играющая программа, при прочих равных условиях, должна будет просматривать игру довольно далеко вперед. Траскотт пишет, например, о программе, разработанной им совместно с Енсеном: «Затрачивая в среднем 5 с на один ход (на машине IBM 370/165), программа играет достаточно хорошо, чтобы выигрывать у авторов. Если на один ход будет затрачиваться в среднем 20 с, то она, вероятно, будет сотым по силе игроком в США. При 80 с на ход программа, наверное, войдет в десятку ведущих игроков мира».

Читая описание игровой программы, в котором раскрывается механизм ее внутренних операций, иногда чувствуешь некоторое разочарование. Когда играешь против программы, не зная ничего о том, как она работает, нетрудно наделить ее замечательными «интеллектуальными» способностями, которых на самом деле она вовсе не имеет. Многие любят пофантазировать. Однако можно надеяться, что утеря приятных иллюзий может быть компенсирована восхищением перед изящной структурой и механизмом действия игровой программы.

Мальчик спросил, можно ли написать программу, играющую в шашки, которая никогда не проиграет человеку. Этот вопрос затрагивает две центральные проблемы создания искусственного интеллекта: во-первых, *что такое* интеллект и, во-вторых, *в какой степени* программы могут проявлять свойства, присущие интеллекту. Когда мы фокусируем такие вопросы на ограниченные проявления человеческого интеллекта, например в играх, они, кажется, растворяются в бесчисленных технических деталях игры. Может быть, в этом и заключается ответ?

Поиски в области теории искусственного интеллекта продолжаются. Некоторые исследователи считают, что изобретение думающей машины в каком-то смысле напоминает проблему изобретения летающей машины. Возможно, появится что-то вроде теории «интеллектуального крыла». Чтобы летать, не обязательно сооружать механическую птицу — для этой цели вполне подходит самолет. Сможем ли мы добиться того же в области искусственного интеллекта?

Книги

Зеленые удобрения; законы физики и спорт; индейцы Северной Америки; радиоактивность окружающей среды

ФИЛИП МОРРИСОН

ТЕХНОЛОГИЯ БИОЛОГИЧЕСКОЙ ФИКСАЦИИ АЗОТА И СЕЛЬСКОЕ ХОЗЯЙСТВО ТРОПИКОВ. Под редакцией *Питера Х. Грэхема и Сьюзен С. Харрис*

BIOLOGICAL NITROGEN FIXATION TECHNOLOGY FOR TROPICAL AGRICULTURE, edited by Peter H. Graham and Susan C. Harris. Centro Internacional de Agricultura Tropical, Cali, Colombia. Available in the U.S. from UNIPUB, 345 Park Avenue South, New York, N.Y. 10010 ($35)

ГЕННАЯ ИНЖЕНЕРИЯ РАСТЕНИЙ (Отчет о конференции). Составитель *Лесли Робертс*

GENETIC ENGINEERING OF PLANTS, a conference summary by Leslie Roberts. Board on Argiculture, National Research Council. National Academy Press, Washington, D.C. 20418 ($9.50)

Невысокие злаки с тяжелыми, налитыми золотым зерном колосьями занимают сейчас в мире огромные площади от мексиканских, египетских и пенджабских пшеничных полей до рисовых плантаций южного Китая. И повсюду рядом с ними вырастают высокие башни новых заводов по производству аммиака — сейчас они есть и в мексиканском штате Чиуауа, и в индийском округе Барода, и в южнокитайском городе Чанша, и во многих других местах, где раньше и не слышали о химии. В полной мере используя достоинства новых сортов растений, удобрения и системы орошения, умудренные опытом крестьяне за последние 25 лет совершили свою «зеленую революцию». Образно говоря, жестокий Князь Голод уже изгнан, и по-видимому навсегда, из городов и основных сельскохозяйственных районов, теперь он совершает лишь короткие набеги во время засух, например в Африке и Бразилии. Однако своевольный Король Бедность до сих пор царит в большинстве своих владений — везде, где люди еще вынуждены вести хозяйство на небольших или малоплодородных участках земли.

Несколько лет назад в мире началась новая кампания. Ее цель — сделать современные агрономические достижения доступными даже для тех, кто слишком беден, чтобы позволить себе артезианские колодцы и азотные удобрения, без которых невозможно эффективно использовать новые высокоурожайные сорта. Книга Грэхема и Харрис вышла в свет в Кали, одном из центров международной сети научно-исследовательских сельскохозяйственных учреждений, которые служат своего рода арсеналами зеленой революции, представляет собой отчет о недавно проводившейся здесь конференции. В книге собраны краткие разнообразные по тематике сообщения примерно 200 участников встречи из 30 с лишним стран. Начавшуюся кампанию, о которой идет речь, можно назвать «сине-зеленой реформой», поскольку ее задача — изыскать возможность делать в более широких масштабах то, что́ сине-зеленые водоросли (цианобактерии) с незапамятных времен делали для человека на заливаемых рисовых полях.

Практические результаты впечатляют. Один из наиболее популярных примеров приводят исследователи из Института по изучению риса, расположенного у подножия высокого вулканического конуса горы Майон на юге острова Лусон (Филиппины), — с местной лаборатории в свое время началась вся существующая ныне система научно-исследовательских сельскохозяйственных центров. Агрономы тщательно измеряли азотный баланс нескольких опытных рисовых делянок на протяжении 23 последовательных урожаев. Оказалось, что считающийся высоким урожай в 34 центнера риса с гектара держался на них годами без внесения в почву дополнительного азота. Учитывая поступление свободного азота с дождевой и оросительной водой, можно подсчитать, что в образовании каждого урожая риса участвует приблизительно 56 кг/га связываемого азота — примерно такое его количество потребовалось бы вносить в виде химических удобрений. Пятьдесят лет назад было установлено, что азот переходит в усваиваемую растениями форму благодаря деятельности свободноживущих азотфиксирующих бактерий, во множестве населяющих затопляемые земли. В настоящее время из зерновых культур только рис может пользоваться их «услугами». Конечно, при внесении в почву азота часто бывают еще более высокие урожаи.